Three-dimensional image analysis and display by space-scale matching of cross sections

V. R. Algazi, B. W. Reutter, W. L. G. van Warmerdam, and C. C. Liu

Graphics, Image, and Vision Engineering Research Laboratory, Center for Image Processing and Integrated Computing, University of California, Davis, Davis, California 95616

Received September 13, 1988; accepted February 14, 1989

We present a framework and a set of techniques for the analysis and display of three-dimensional experimental data or images. We assume that the data are available in the form of two-dimensional cross sections of the three-dimensional data set. We describe our approach, which has for goals to extract significant information from the three-dimensional data set and to display this information as objects that can be manipulated in three-dimensional space. The high-contrast transitions of two-dimensional cross sections are extracted first; they define a set of contours to be matched from cross section to cross section. This matching is performed by space-scale analysis of the orientation of contours on adjacent cross sections. By modeling the contours as B splines, we then make use of three-dimensional B-spline patches to generate significant surfaces that can be displayed, rendered, and rotated with standard computer graphics techniques and specialized processors.

1. INTRODUCTION

In this paper we consider the analysis and display of three-dimensional (3-D) objects, images, or data sets. Some techniques for the determination of physical attributes of objects and organisms in 3-D space are now well established. Noteworthy methods include physical tomography, in which objects are actually sectioned; computer-assisted tomography, in which sections are determined by computation from projection images; and optical sectioning, in which the image plane of a microscope with a small depth of focus is stepped across a semitransparent object or organism.[1-3] Other inverse techniques suited to different modes of propagation and sensing, such as seismic waves and ultrasound, also result in estimates of the spatial attributes of 3-D objects.[4,5] A closely related area of interest is the computer modeling and analysis of materials and objects in three dimensions, commonly performed by finite-element analysis, in which the data consist of the results of a computer simulation. In such cases, often one must analyze and display massive sets of 3-D data in order to refine the models, to understand their implications, and to make use of their results.[6,7]

At the present time, the approach that is used most commonly to analyze and display the data is to obtain first a sparse subset of data by selecting voxels with a density value within a selected narrow range. If the experimental data are calibrated well, then such a selection process will result in meaningful spatial groupings, which are then displayed.[8] This is sufficient, for instance, in x-ray tomography, to discriminate bone and cartilage broadly from soft tissue. However, it is well known that low-spatial-frequency radiometric errors are prevalent in most imaging systems and that simple thresholding is not adequate for feature selection. Further, attributes of interest may be related not only to density but to differential properties or to gross or local morphology. In this paper we describe a framework for analysis that draws on recent approaches and results of image analysis, computer vision, and computer graphics used jointly for 3-D image analysis and visualization.

The questions to consider first for this analysis problem are: What is the important information contained in a 3-D object, and how can it be captured, structured, and visualized? Without needing to refer to a specific field of application, one can state generally that the information of interest is contained in the spatial structure of subregions that differ substantially from their surrounds, i.e., in the shape, structure, and extent of anomalies. Because we may not know, at the outset, the spatial extent of the anomalies, we assume that the sample images, which are here the cross sections of the object, are of a sufficient resolution that the cross sections of such anomalies can be extracted and described effectively. It is seldom the case that a high density of image cross sections is necessary as well, i.e., that a uniformly sampled 3-D array of data is needed, because anomalies of some spatial extent can be detected and can be described fairly well by successive object cross sections that are several pixels apart. The computational difficulties and the cost of obtaining cross sections also encourages us to make use of as few such sections as possible. Note that we do not consider here the significant imaging and processing problems that arise in obtaining good cross sections.[9] Thus, given a limited number of cross sections, we then address the problem of constructing the 3-D structures of the objects under study. In the approach that we present in this paper, we first extract significant data transitions in two dimensions. For us, these features are the major transitions within each cross-sectional image, each characterized possibly by several parameters, such as width and shape, as well as by their locations. Once these features are determined, we must match them from section to section to generate a sampled representation of the significant 3-D surfaces.

For these 3-D surfaces, the extensive techniques and tools developed in recent years for computer graphics can then be used to construct the surfaces from their samples, to control

0740-3232/89/060890-10$02.00

the object orientation in 3-D space, and to choose alternatives for the rendering and viewing of these experimental data surfaces. The overall processing and display then allow analysts to interact with and to gain insight into specific problems by viewing these constructed images, which represent quantitative parameters, visible significant features, and their spatial relationship.

2. EXTRACTION OF SIGNIFICANT FEATURES IN TWO DIMENSIONS

In order to use a common terminology for all applications, we shall refer to the quantitative data of interest as image intensity. Thus we assume that the actual physical value is or has been transformed into a scalar function of two dimensions. The boundaries between image regions of differing intensity characteristics correspond to the boundaries between 3-D regions. In this section we present a method for generating a local description of these boundaries, which we call the intensity transitions, by using the Gaussian smoothed derivatives of the image. Connected local transitions will form contours and provide features for matching with the corresponding boundaries in adjacent section images.

A. Introduction

A classical approach to the extraction of edge or transition information involves the use of Gaussian derivative[10] filters, which are characterized by σ, the filter scale parameter, which affects both the spatial-domain and the frequency-domain characteristics of the filters. In such filters, the data are smoothed by a Gaussian kernel, and then successive derivatives are taken. A specific and useful filter is $\nabla^2 G(x, y, \sigma)$ [Laplacian of a two-dimensional (2-D) Gaussian], which has a radially symmetric impulse response. The line response of $\nabla^2 G(x, y, \sigma)$ is $g_2(x, \sigma)$, the second derivative of a one-dimensional (1-D) Gaussian. The $g_2(x, \sigma)$ is sensitive to extrema in the slope of the intensity and responds with a zero crossing at those locations. A significant feature of the filter response is the slope at the zero crossing, which is called the zero-crossing slope. In 2-D filtering with the $\nabla^2 G(x, y, \sigma)$, the zero-crossing slope is evaluated in the direction of the gradient of the response at the zero crossing.

Studies of Gaussian-derivative-based edge detection arose from studies of human visual processing.[11] Marr proposed a model of multiple-resolution edge detection in which the image is filtered with a set of $\nabla^2 G(x, y, \sigma)$ filters covering a range of scales. In the model it is postulated that the edge description can be obtained from the locations of the zero crossings and the values of the zero-crossing slopes in the response of the $\nabla^2 G(x, y, \sigma)$ filter. We have developed[12,13] an algorithm, which we describe briefly here, for obtaining such a multiple-resolution edge description from the zero crossings and slopes. This problem, which is of great importance to the analysis of multidimensional images and data, was not addressed directly in numerous recent contributions on edge extraction.[14,15]

A critically important feature of our algorithm is its attention to the width of the intensity transitions in an image. (The width of a transition is the distance over which a significant portion of the intensity change occurs). The width of an image transition, a variable bounded below by the blur of the imaging system, is a significant parameter for two reasons. First, knowledge of the transition width results in a richer description of the transition and allows us to model a progressive spatial change of data characteristics. Second, an accurate average description of the transition location and shape is obtained at the scale of the transition, i.e., at a σ value where the support of the Gaussian smoothing matches the transition width.

Using a ramp model of image edges, our algorithm estimates the scale $\hat{\sigma}$ of a transition from the zero-crossing slope in the second-Gaussian-derivative response to the edge. Estimates of the transition location and shape are obtained from the zeroth, first, and third Gaussian derivatives of the edge evaluated at the scale $\hat{\sigma}$.

We shall describe the use of this algorithm both for extraction of image transitions and for matching contours among image sections. Because of these two applications, both 1-D and 2-D versions are important to our applications. We first develop the algorithm for detecting and describing isolated 1-D transitions, and then we apply the algorithm to images. Finally, we discuss the features obtained from the transition descriptions.

B. Characterizing an Isolated One-Dimensional Transition

We now present an algorithm for detecting and describing an isolated 1-D transition from knowledge of the derivatives of the Gaussian smoothed edge as a continuous function of σ.[16]

We first model an isolated 1-D edge by a ramp transition of width α between regions of constant intensity:

$$s(x) = \begin{cases} 0 & x \le -\dfrac{\alpha}{2} \\ \dfrac{1}{2} + \dfrac{x}{\alpha} & -\dfrac{\alpha}{2} \le x \le \dfrac{\alpha}{2} \\ 1 & x \ge \dfrac{\alpha}{2} \end{cases}. \tag{1}$$

Transitions are detected from the zero crossings in the $g_2(x, \sigma)$ response to the edge. It can be shown that the response of the second derivative of a 1-D Gaussian filter $g_2(x, \sigma)$ to the edge $s(x)$, denoted $sg_2(x, \sigma)$, contains a zero crossing at the transition center ($x = 0$) for all σ:

$$sg_2(0, \sigma) = \int_{-\infty}^{\infty} s(t - x) g_2(x, \sigma) \mathrm{d}x \big|_{t=0} = 0. \tag{2}$$

We estimate the transition scale (the σ value at which the support of the Gaussian smoothing matches the transition width) from the zero-crossing slope of the $g_2(x, \sigma)$ response to the edge. The σ^2-weighted zero-crossing slope, $\sigma^2 sg_3(0, \sigma)$, in the ramp response above is given by

$$sg_3(0, \sigma) = -\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\alpha^2}{8\sigma^2}\right). \tag{3}$$

We have shown that the magnitude of the expression above attains a maximum with respect to σ at $\sigma = \alpha/2$.[13]

Considering transitions that are not ramps, we have also shown that the zero-crossing slope of the response to a large class of edges attains a maximum with respect to σ at a scale $\hat{\sigma}$ related to the transition width. We call this $\hat{\sigma}$ the estimate

of the transition scale. For other edges besides ramp transitions, we have approximately $\hat{\sigma} = \alpha/2$.

The estimate of the transition location is $\hat{x}$, the location of the $g_2(x, \sigma)$ response zero crossing obtained at the scale $\hat{\sigma}$. The zero crossing obtained at the scale $\sigma = \hat{\sigma}$ is an excellent estimate of the transition location.[13] The accuracy in locating the transition can be evaluated for a ramp edge in white Gaussian noise. It can be shown that an estimate of the standard deviation σ_{zc} of the location of the zero crossing is[13]

$$\sigma_{zc} = \frac{1}{\mathrm{SNR}_{\mathrm{in}}} \left(\frac{3\sqrt{\pi}\sigma}{16}\right)^{1/2} \exp\left(\frac{\alpha^2}{8\sigma^2}\right), \tag{4}$$

where $\mathrm{SNR}_{\mathrm{in}}$ is the edge contrast divided by twice the standard deviation of the noise. The expression above attains a single minimum at the scale $\alpha/\sqrt{2}$, which is slightly larger than $\hat{\sigma}$. However, since σ_{zc} is a broad function of σ, the zero crossing obtained at $\hat{\sigma}$ is still a good estimate of the ramp location.

We model the transition shape further with a cubic polynomial about $\hat{x}$ composed of the Gaussian derivatives of the edge evaluated at the scale $\hat{\sigma}$:

$$s(x) \approx sg_0(\hat{x}, \hat{\sigma}) + \left(sg_1(\hat{x}, \hat{\sigma}) - \frac{\hat{\sigma}^2 sg_3(\hat{x}, \hat{\sigma})}{2}\right)(x - \hat{x}) + \frac{sg_3(\hat{x}, \hat{\sigma})}{6}(x - \hat{x})^3, \tag{5}$$

where $sg_0(\hat{x}, \hat{\sigma})$, $sg_1(\hat{x}, \hat{\sigma})$, and $sg_3(\hat{x}, \hat{\sigma})$ are the zeroth, first, and third Gaussian derivatives of the edge evaluated at the scale $\hat{\sigma}$ and the location $\hat{x}$. Thus we can estimate the shape of an arbitrary transition by determining its width and then by estimating the coefficients of the cubic polynomial above.

The procedure described above extends the studies of Marr and Hildreth by using the zero-crossing slope (third Gaussian derivative) to detect transitions and to estimate their width. However, the weighted first Gaussian derivative can also be used. For the ramp edge [Eq. (1)], the first Gaussian derivative evaluated at the transition center is

$$sg_1(0, \sigma) = \frac{2}{\alpha} \int_0^{\alpha/2} \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{\frac{-x^2}{2\sigma^2}\right\} dx. \tag{6}$$

When weighted by roughly $\sigma^{2/3}$, Eq. (6) attains a maximum at $\hat{\sigma} = \alpha/2$. In fact, the first Gaussian derivative response, when weighted by any power of σ between 0 and 1, attains a maximum for a large class of transitions. (For ramp transitions, as well as many other types of transition, the magnitude of the first Gaussian derivative response is an increasing function of σ if weighted by σ. Though a weight of σ results in a simple relationship between the responses for transitions of different widths,[17] such a weight cannot be used to estimate the width directly.)

C. Two-Dimensional Algorithm for Images

The approach described here[12] is an extension of the 1-D algorithm and works best for transitions that are isolated from one another.

In two dimensions, the algorithm uses the 1-D results by dividing connected zero crossings appearing in the $\nabla^2 G(x, y, \sigma)$ filter outputs into local segments.[13] The segments, characterized by a location, a phase [direction of the $\nabla^2 G(x, y, \sigma)$ gradient at the zero crossing], a zero-crossing slope, and zeroth and first derivatives, are treated much as 1-D zero crossings. This 1-D description of local zero-crossing segments is simple but gives a satisfactory description of the intensity transitions with extents longer than their widths.

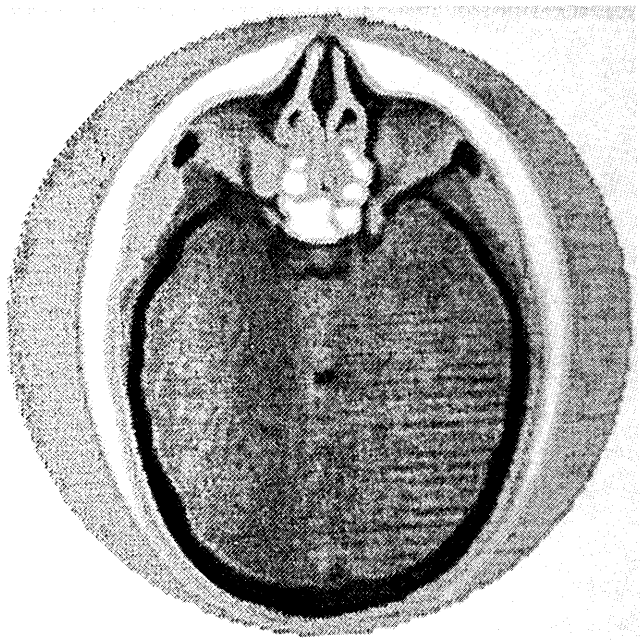

Fig. 1. Computer-assisted tomography scan of a human brain.

The algorithm examines the zero-crossing slopes of spatially adjacent segments in neighboring channels for a discrete local maximum signaling the existence of a candidate intensity transition at the channel scale. The algorithm describes transition candidates with the parameters characterizing the segment with the local maximum slope.

An analysis of the effects of additive white Gaussian noise in an image predicts that the descriptions obtained for narrow transitions are most sensitive to noise. For a ramp transition of width $\alpha = 4$, the algorithm obtains an accurate description when $\mathrm{SNR}_{\mathrm{in}}$ is 5 or more.[13]

D. Examples

We have performed a three-channel version of our algorithm (σ of 2, 4, and 8 pixels) on a special-purpose Gould image processor. Figure 2 shows an example of the algorithm output for the image in Fig. 1. Shown are the location estimates of the transitions in the image and the magnitude of the zero-crossing slopes. The contributions from the various channels can be distinguished by the printed width of the line segments. The widest segments come from the 8-pixel channel; the narrowest come from the 2-pixel channel. The darkness of the segments is proportional to the magnitude of the zero-crossing slope. Notice that the results contain some transitions detected by the 4- and 8-pixel channels.

E. Transition Contours

The local descriptions of adjacent transitions are connected to form contours whose characteristics yield two useful feature sets for matching. The first set is the description of the intensity transitions obtained by using the procedure summarized in Subsection 2.C. The second feature set is the description of the contour shape, obtained by applying the 1-

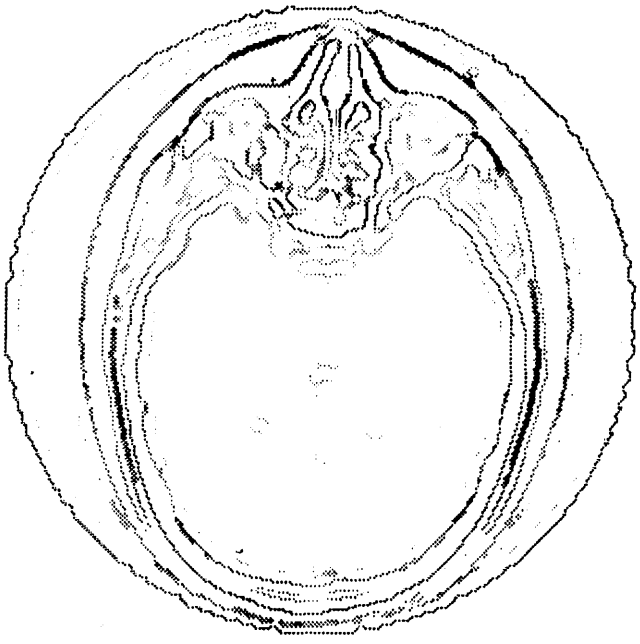

Fig. 2. Location estimates of the intensity transitions in Fig. 1.

D algorithm of Subsection 2.B to the 1-D waveform obtained by plotting the contour orientation as a function of the arc length. When represented in this form, the contour shape may be analyzed in terms of its transitions in orientation space.

In Section 3 we describe a procedure for matching corresponding contours in adjacent cross sections, using only the description of the contour shape. For simplicity, we do not extract the first feature set. (The use of more-complete transition information is discussed further in Section 6.) The contour locations are obtained by filtering the image at a single σ value. Note that the zero crossings of $\nabla^2(G(x, y, \sigma)$ form contours without any gaps. Thresholding based on the magnitude of the zero-crossing slope will not significantly affect continuity for the transitions that we have considered in our examples.

3. MATCHING CONTOURS AND FEATURES ACROSS SECTIONS

As we have discussed in Section 2, the features extracted from image cross sections are significant transitions of the 2-D data. Thus a selection has already been made, at this stage in the analysis, of properties of the data that may form a 3-D structure of interest within the entire object or data set. The simplest way to generate such 3-D structures is to think of them as surfaces specified by their cross sections. Thus one contour or a set of contours for these surfaces has been determined from each section. For sharp, localized transitions, the location of the transition, or image edge, is well defined and is thus an obvious choice for a contour. For extended or wide 2-D transitions, a contour, which is a section of the 3-D surface, is obtained by choosing, for instance, the midpoint of the transition.

A. General Background on Contour Matching

The applications of contour matching that are considered most commonly in the technical literature are for artificial objects within a scene, which result in images to be matched that exhibit rotations, translations, scale changes, and partial occlusions. These specific problems do not occur in the case of adjacent parallel sections, since an approximate alignment of cross sections is readily available and the scale is uniform. However, contours change from section to section, and thus the detailed matching is more difficult and more subtle. We shall use the approach of matching at different resolutions, or in scale space, first to obtain an approximate match and then to complete detailed matching by using continuity and smoothness constraints along a contour.

B. Characterization of Contours

Contours are characterized best as a function of the arc length. The origin, for contours roughly concentric, is chosen with reference to a single point in each image at a common polar angle for all images. Useful position and shape descriptors are the orientation and the curvature of the contours as well as the contrast and the width of the image transitions from which the contours have been extracted. Because the geometry of the sectional images has been calibrated, measured, and corrected, we can perform a gross matching and alignment of sectional contours simply by matching the polar angles of contour segments. The examples of contours shown in Fig. 3 demonstrate some of the features of the matching problem. The gross matching is reasonably easy, but detailed features do not match.

C. Contour Matching

The objective of contour matching is to establish detailed alignment between pixels on the corresponding contours extracted from adjacent cross sections. Ultimately, a warping (mapping) function between the arc lengths of the two contours is computed. Our matching algorithm is based on the scale-space, or resolution-dependent, description of waveforms introduced by Witkin.[18] Each contour or portion of a contour is first transformed into a 1-D orientation function of the arc length. Examples are shown in Fig. 4. Thus the problem has now been reduced to a single dimension. The 1-D orientation function is filtered by a set of $g_2(x, \sigma)$ operators at different resolution scales, as discussed in Section 2 above. For a set of scales σ, zero crossings are extracted from the filtered output. These zero crossings, plotted as a function of the scale σ, form the pattern shown in Fig. 5, known as the fingerprint of the specific 1-D function. For a contour, the fingerprint shows the location of significant transitions of the orientation as a function of the spatial scale, σ. Figure 5 shows that the fingerprints for adjacent cross-sectional contours match quite well for large σ. Since the 1-D function is the orientation as a function of arc length, the significant transitions are the local high-curvature regions, or rapid changes in orientation, with a contrast that is the total change of orientation in the transition.

The matching process starts at a large σ. For that large σ, the fine features of the contours have been smeared by filtering, and potential matching candidates from section to section are determined by the polar angles, or even by the length of contours, at which zero crossings occur. The sign of the 2-D image-transition slope can also be used as a selecton criterion in matching. This simple preprocess may give

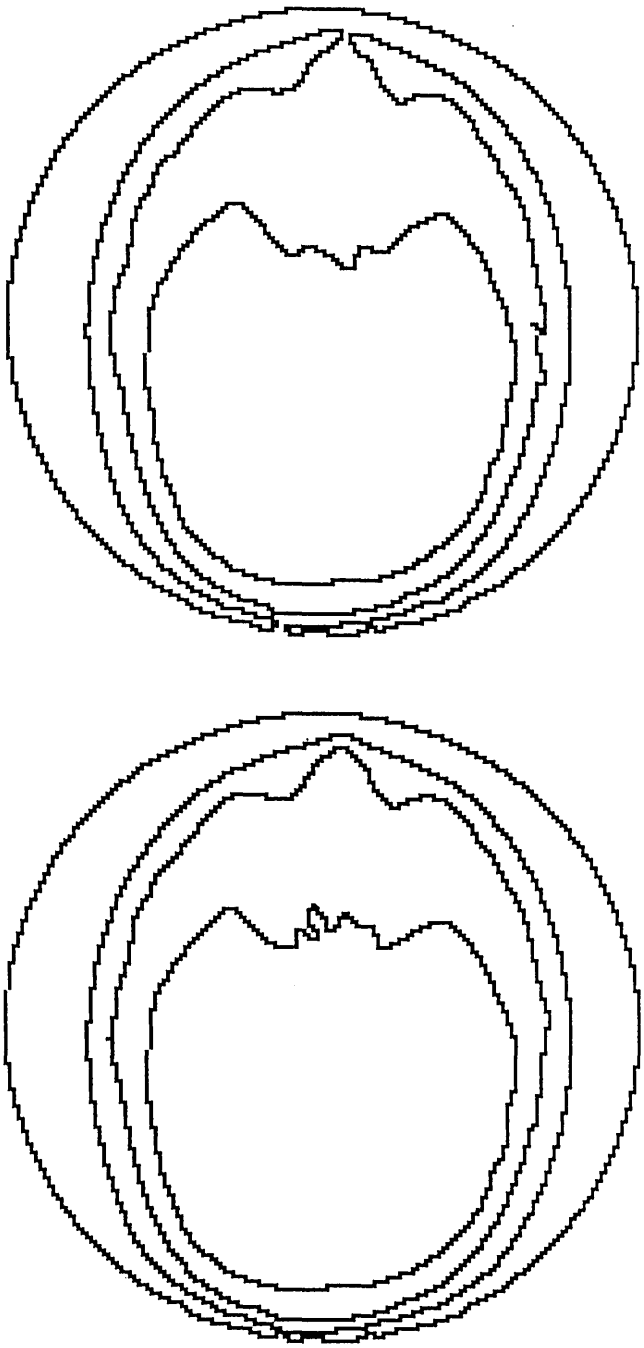

Fig. 3. Contours in adjacent cross sections.

good initial results if the potential match candidates are sparse in a neighborhood.

D. Matching Procedure

1. First Pass: Isolated Transitions

For each of the contours we have available the fingerprint, and for each point on a fingerprint branch we have the strength [the magnitude of the σ^2-weighted zero-crossing slope in the $g_2(x, \sigma)$ response] of the transition in the orientation function. As discussed in Section 2, the maximum strength along a branch, i.e., as a function of σ, indicates the width of the transition. For isolated transitions, the fingerprint exhibits a single, roughly vertical branch. The best estimate of the location of the midpoint of the transition is obtained for the σ that maximizes the strength. However, the transition can be localized well even for a mismatched and fairly large sigma. If we substitute $\alpha = 2\sigma$ into Eq. (4), the localization standard deviation σ_l is bounded by

$$\sigma_l \leq \frac{\sqrt{\sigma}}{\mathrm{SNR}_{\mathrm{in}}}, \tag{7}$$

where the transition width is less than 2σ and $\mathrm{SNR}_{\mathrm{in}}$ has been defined in Section 2. For a high-strength, rapid transition, or one that approximates a ramp well (i.e., $\mathrm{SNR}_{\mathrm{in}} > 3$), then $\sigma_l \leq 1$ for $\sigma = 9$. This result indicates that noise-free, isolated transitions can be localized well for sigmas as large as 8 or 10.

Thus, after the fingerprints of possible matching contours are obtained, our algorithm compares these fingerprints for all sigmas larger than a predetermined space scale (say, $\sigma_p = 10$). The locations along fingerprints where the strength is maximum are determined for all σ values larger than the space scale σ_p along each fingerprint branch. For transitions of sufficient strength, the maxima and their locations are used to match with the fingerprints for adjacent sections. For isolated transitions, as discussed above, a value of $\sigma \approx 10$

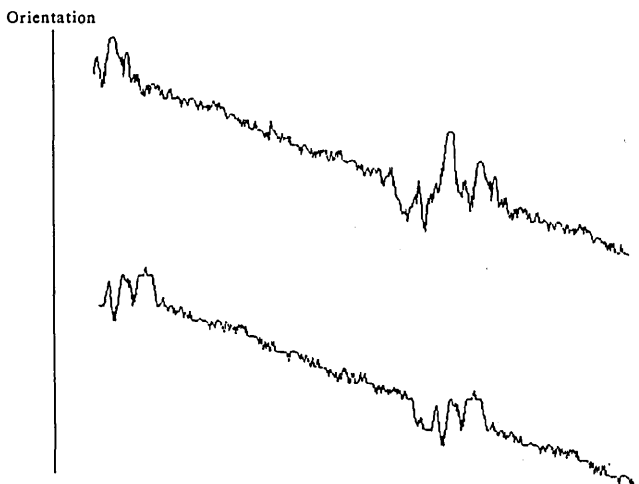

Fig. 4. Orientation functions for matching contours.

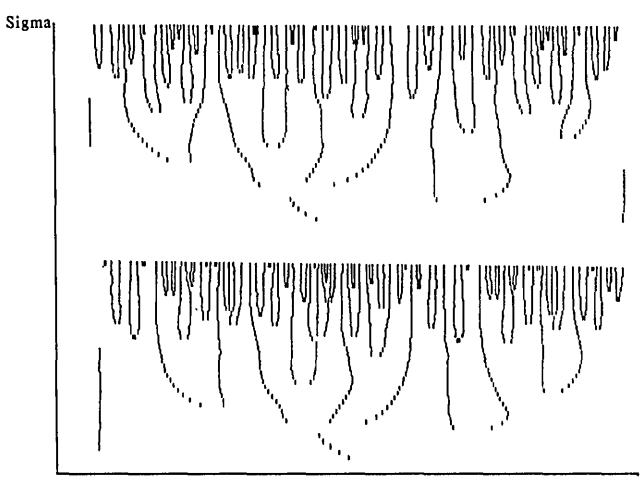

Fig. 5. Fingerprint diagrams for Fig. 4.

provides a good localization of the transition, and the process can be stopped.

Note that results from this first step matching should be consistent with our preliminary alignment assumption, and thus only a relatively local region is considered in matching each fingerprint contour.

2. Decreasing Sigma and Multiple Transitions

As sigma decreases below the limit value of $\sigma_p = 10$, existing fingerprint branches extend, and new branches may appear. New maxima of the Gaussian derivative strength, detected along each branch, define new transitions or may update previously detected transitions at a smaller matching scale. They may be used for a more detailed match between cross sections.

Three types of fingerprint pattern are identified:

(1) A single isolated transition and therefore a fingerprint with no branching.

(2) Multiple transitions. They do not overlap one another (i.e., they are not within $\pm 2\sigma_i$ of one another, where σ_i is the matching scale of each transition).

(3) A composite transition, in which multiple overlapping transitions are detected.

The issue is to determine whether we should make use of detailed matches in all cases. For cases (1) and (2), additional matches at finer scales are justified. However, for composite transitions, case (3), matching may be less reliable, since the transitions are not isolated, and it is not necessary because the detailed transitions are close to one another. Therefore no detailed matching is performed, and the previously obtained larger-sigma match is used.

When a detailed match is performed, the procedure is as follows, for a match pair at a coarse level with strengths $S_{1,i}$ and $S_{2,i}$ at $x_{1,i}$ and $x_{2,i}$:

(1) Matching isolated transitions. We use the updated matching scale sigmas and strengths $\hat{S}_{1,i}$ at $\hat{x}_{1,i}$ and $\hat{S}_{2,i}$ at $\hat{x}_{2,i}$, and we check for (a) comparable strengths $\hat{S}_{1,i}$ and $\hat{S}_{2,i}$ (within a ratio of 1.5 to 1) and (b) comparable matching scale sigmas $\hat{\sigma}_{1,i}$ and $\hat{\sigma}_{2,i}$ (within a ratio of 1.5 to 1).

(2) Matching multiple nonoverlapping transitions. We use the updated matching scale sigmas and strengths and we check for (a) comparable strengths, (b) comparable matching scale sigmas, and (c) a consistent order of multiple matches for increased resolutions along each of the two matching contours.

E. Using the Matching Locations

The results of the fingerprint contour matching are a set of matched points along the two corresponding 2-D contours. We refer to these initial matched points as critical points. These critical points may not be sufficient for the determination of the entire warping function. The warping function is taken to be piecewise linear in that the arc lengths are in a linear correspondence between these matching points. Table 1 lists the initial matched points obtained for the pair of contours in Fig. 3 that correspond to the outer surface of the skull.

The approach works well for whole contours or even segments of contours taken tens of pixels apart. Difficulties occur for small features and for image sections near the extremes of closed surfaces.

Note that, for gaps in one of the matching contours, our use of B-spline approximations, as discussed in Section 4, provides a smooth interpolation of the available data.

Table 1. Initial Matched Points Obtained for the Pair of Contours in Fig. 3 That Correspond to the Outer Surface of the Skull

Image	Pixel x	Pixel y	Polar Angle (deg)	Contour Orientation (deg)	Signed Curvature
Top	170	93	128	237	−1
Bottom	169	92	127	243	−1
Top	168	88	133	240	1
Bottom	167	87	132	237	2
Top	145	75	158	272	−4
Bottom	144	75	157	284	−4
Top	79	119	258	12	4
Bottom	80	119	256	9	1
Top	79	123	263	6	−1
Bottom	79	124	263	6	−1

4. GENERATION OF SIGNIFICANT SURFACES IN THREE DIMENSIONS

We study now the interpolation of the matched sets of points of the 2-D contours to generate 3-D surfaces.

The 3-D surfaces are generated by selecting networks of points on the sets of matched contours and by interpolating these networks of points with surface patches. There are several advantages to this approach. First, significant computational savings are achieved by having to store and manipulate only these networks of points and not the entire set of cross-sectional images. Second, the generation and manipulation of surfaces by computer graphics algorithms and the corresponding specialized software and hardware provide us with a powerful set of concepts and tools. Among the 3-D-surface generation techniques, we have chosen to represent the surfaces in terms of B-spline patches. B splines have the desirable property of yielding a parametric, coordinate-free representation that is invariant under translation, rotation, or scaling of the coordinate system.[19] Thus, once such a B-spline surface representation has been found, the manipulation of data and surfaces in 3-D space requires only limited computations and is supported by specialized hardware. However, the B-spline representation depends on control vertices that are not, in general, on the B-spline surface. Thus we must determine the needed control vertices from our experimental data points. We have used the Barsky–Greenberg algorithm, which calculates the B-spline representation, i.e., the control vertices, efficiently for a surface that interpolates a network of data points.[20,21]

Further, in order to obtain this B-spline surface representation we must have a set of points that can be organized as a rectangular graph. For the case of a set of cross-sectional contours, this requires the selection of the same number of points on each contour and the establishment of the one-to-one correspondence between points on adjacent contours.

Thus the number of points selected on each contour will depend on the minimum number of points that are required

to generate a good 2-D B-spline representation for the most complex contour.

A. Selection of Additional Matching Points on Contours

To establish the one-to-one correspondence between points on adjacent contours, we begin with the sets of critical points obtained from the fingerprint contour matching discussed in Section 3. These critical points delimit contour segments on which additional points are selected, so as to increase the density of points on the contours.

For a critical point $\mathbf{P}$ that has been matched with a point $\mathbf{P}_a$, say, on the contour above but not with a point on the contour below, a corresponding point $\mathbf{P}_b$ on the contour below is selected so that the set of points can be organized as a rectangular graph. Using a linear warping function determined by the critical points that bracket $\mathbf{P}$ and are matched to the contour below, the point $\mathbf{P}$ is mapped to a point $\mathbf{P}_b$ on the corresponding contour segment below. Similarly, this point $\mathbf{P}_b$ is mapped from its contour segment to the corresponding contour segment below, and so on.

After this process is completed, we have a rectangular graph of points that delimit families of matching contour segments across more than two contours. For a low-curvature contour segment it seems reasonable to select additional points that are spaced evenly along the segment, subject to some maximum permitted arc length. To increase the density of points, each family of contour segments is first subdivided as needed into families of subsegments so that the longest arc length in each family of subsegments does not exceed the maximum permitted arc length. The subsegments of a given segment have equal arc lengths, which do not exceed the maximum permitted arc length.

Another useful criterion is to consider the integrals of signed curvature along the arcs. Integrating signed curvature along a contour is equivalent to keeping track of the net change in the orientation of the unit tangent vector to the contour, as one traverses the contour. If the orientation of the unit tangent vector changes by a large increment, the subsegment is further subdivided.

Together, the maximum-arc-length and orientation-increment criteria permit a selection of points that is driven by the complexity of the contour, while maintaining a minimum level of resolution for simple contours. Thus the number of additional points selected on the contour segments in a family is determined by the number of additional points that are needed to represent the most complex segment in the family. The result is a network of contour points, which we must interpolate with uniform bicubic B-spline surface patches.

B. Matching Cross Sections for Partial Contours

The matching of partial contours requires more discussion. Recalling the several criteria that we have used for matching, they are (1) the use of polar angles for a gross determination of matching regions along contours, (2) the use of critical points on the contours, determined by the scale-space matching algorithm, and (3) the maximum-arc-length and orientation-increment rules. For partial contours, the polar-angle correspondence can still be used to perform an approximate match. In order to perform a more accurate match, we first make the observation that the matching of the portions of contours between two successive matched critical points does not require any special provisions for partial contours; that is, the matching rules are based on information from adjacent critical points and do not require a closed contour. Thus the problem for partial contours occurs at the end of the partial contour, beyond the last critical point and for the case in which there are no critical points at all. This is an issue that we did not encounter in our examples and that will require additional study.

C. Computation of Control Vertices from Data

Denote the network of $m \times n$ contour points by the set of position vectors

$$P = \{\mathbf{P}_{i,j} | i = 0, \ldots, m-1; j = 0, \ldots, n-1\}, \qquad m, n \geq 4, \tag{8}$$

where m is the number of contours and n is the number of points per contour. The position vector $\mathbf{Q}_{i,j}(u, v)$ for a point on the (i, j)th uniform bicubic B-spline surface patch is simply the weighted average of the position vectors for 16 control vertices:

$$\mathbf{Q}_{i,j}(u, v) = \sum_{r=-1}^{2} \sum_{s=-1}^{2} b_r(u) \mathbf{V}_{i+r,j+s} b_s v, \quad u, v \in [0, 1], \tag{9}$$

where the $b_r(u)$ are the uniform bicubic B-spline basis functions and

$$V = \{\mathbf{V}_{i,j} | i = -1, \ldots, m; j = -1, \ldots, n\} \tag{10}$$

is a network of control-vertex position vectors.

Thus the problem is to determine the unknown control vertices that will generate a surface that passes through the experimental data set P. The Barsky–Greenberg algorithm calculates the network V of control vertices efficiently from the points on the B-spline surface, such that the network P of contour points is interpolated by the corners of the resulting B-spline surface patches. There is continuity of position at each boundary where patches join as well as continuity of the first two partial derivative vectors with respect to the parametric (u or v) direction across the boundary.

Although the position vector $\mathbf{Q}_{i,j}(u, v)$ for a point on a surface patch depends only on a neighborhood of 16 control vertices, control-vertex adjustments propagate, so that moving a single data point or contour point requires that all the control vertices in the set V be calculated again. However, if all of the contour points are translated, rotated, or scaled by the same amount, the control vertices need only to be subjected to this same geometric transformation and need not be calculated again from the Barsky–Greenberg equations.

5. DISPLAY AND VIEWING OF SIGNIFICANT SURFACES

In Section 4 we have discussed the problems and techniques in the generation of surfaces to represent the important 3-D data. Since we have a B-spline representation of the significant surfaces, we can now take advantage of computer graphics techniques to display and interact with these surfaces.[22,23] Our work is performed on a Hewlett-Packard 350 Turbo SRX computer graphics workstation. We use the Starbase Graphics Library, which is a package of graphics

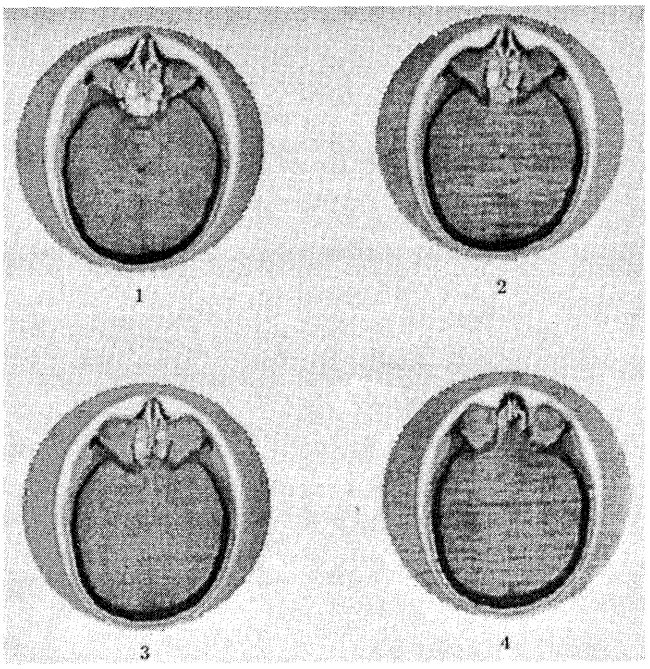

Fig. 6. Four of the eight computer-assisted tomography scan cross sections used to generate the significant surfaces displayed in Figs 7–9.

procedures that are based on the American National Standards Institute's Computer Graphics Interface standard.

Figure 6 shows four of the eight original cross sections used to generate significant surfaces; information extracted is displayed in a number of ways in Figs. 7–9. It is useful to consider these surfaces as actual physical objects. When we examine physical objects, we may move them or bring them to a better light. These are the options that we would like to have for our significant surfaces as well.

Figures 7–9 have been chosen with a number of different viewpoints, to illustrate the need for changes in viewpoint to comprehend the detailed structures of the significant surfaces.

We use simple computer graphics models for the illumination sources in the scene.[24,25] After the illumination sources in the scene have been modeled, the next step is to model the reflection of the incident light from the significant surfaces. In general, light incident upon a surface is partially absorbed and reflected diffusely and partially reflected specularly. Surfaces rendered with diffuse reflection tend to have a dull appearance. Diffuse reflection yields cues about the orientation of the surface with respect to the incident point-source illumination. By interactively moving the surface, viewpoint, and point illumination source with respect to one another, one can explore the fine details of the surface. Specular reflection yields cues about the orientation of the surface with respect to the incident point-source illumination and the viewpoint. Shiny highlights appear when the surface normal vector bisects the angle that is formed by the vector from the surface point to the viewpoint and the vector from the surface point to the point illumination source.

In practice, the B-spline representations of the significant surfaces are approximated by connected polygons. Applying the illumination and reflection models to these polygons, one is likely to perceive the faceted surfaces, unless the

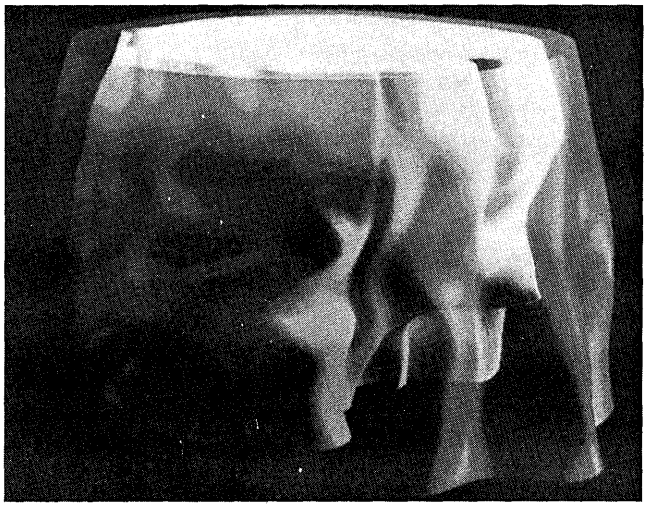

Fig. 7. Use of transparency.

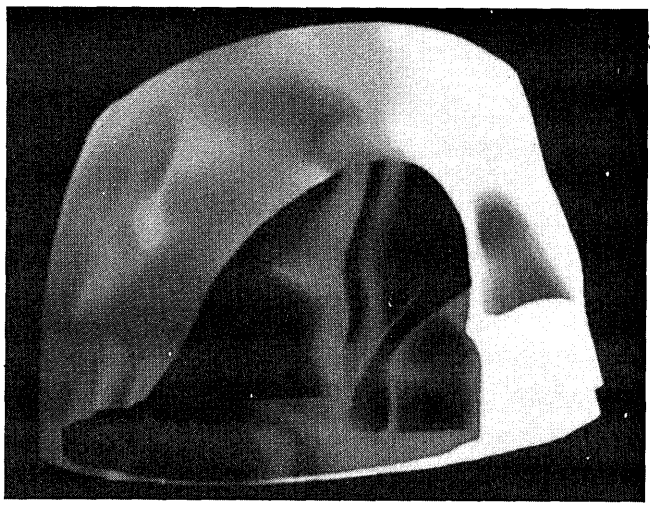

Fig. 8. Clipping of the outer surface to expose the inner surface.

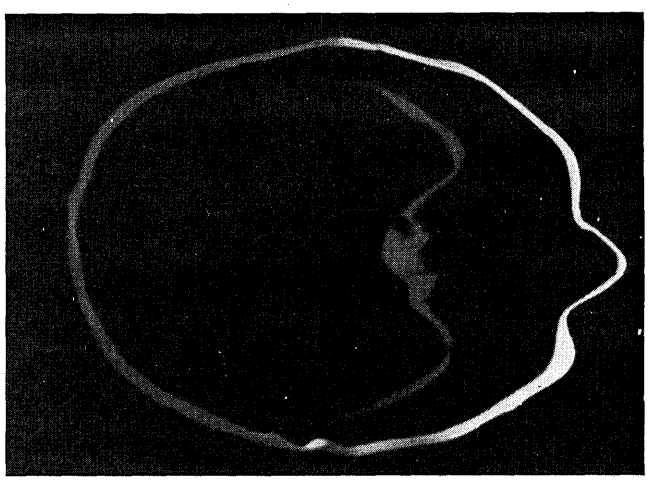

Fig. 9. Use of clipping planes to obtain a tomographic slice.

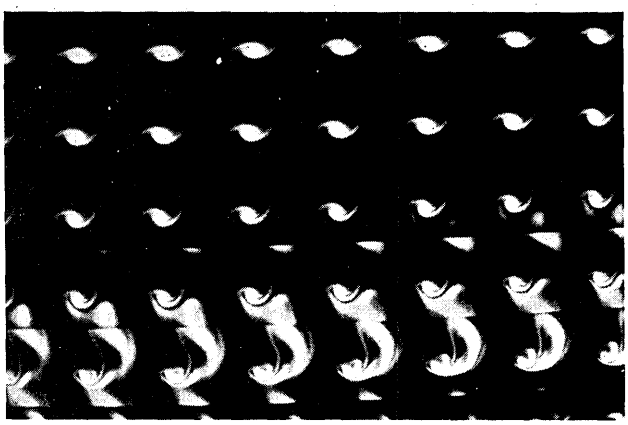

Fig. 10. Cross sections of the three-dimensional vorticity vector magnitude.

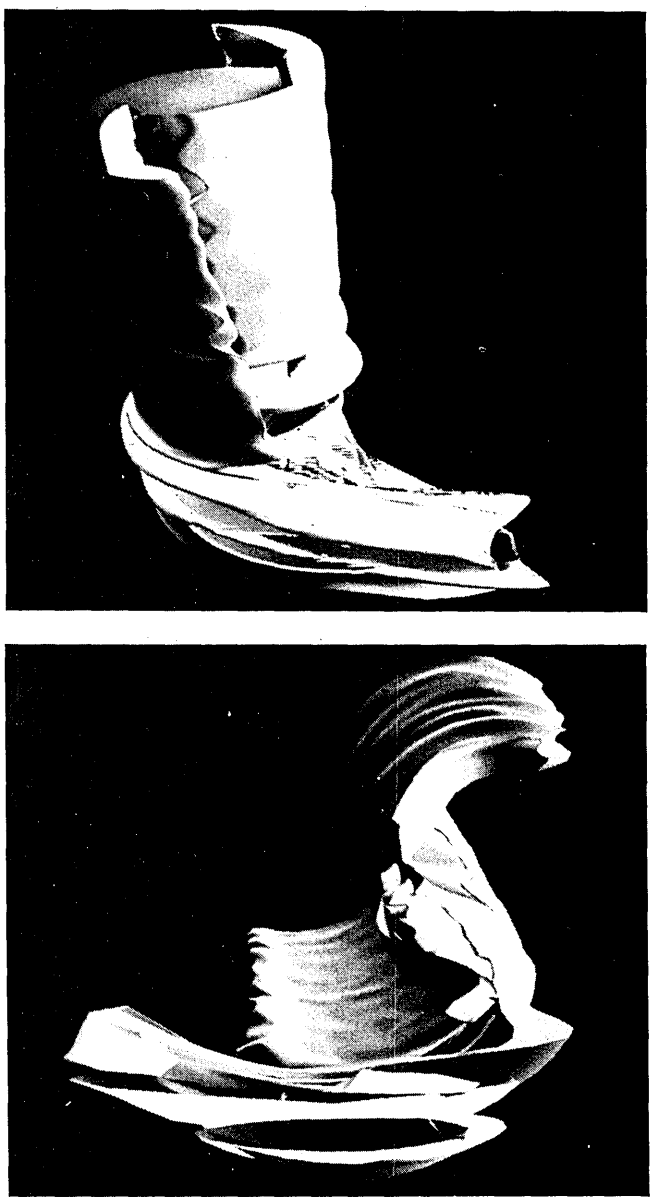

Fig. 11. Surface renditions of the transitions in the data of Fig. 10 (23 cross sections).

polygons are small enough. However, using small enough polygons is likely to be computationally expensive. Two popular shading techniques[25,26] vary the displayed intensity pixel by pixel to smooth intensity transitions between adjacent polygons to produce a good rendering of the significant surfaces with larger polygons.

To gain additional insight into the relationships of the signifiant surfaces with respect to one another, we take advantage of computer graphics techniques that allow us to make a surface transparent or to cut away portions of a surface.

Figure 7 illustrates the use of transparency. In Fig. 8 an inner surface has been made visible by clipping the outer surface with a plane that is perpendicular to the line of sight. Figure 9 illustrates that, by defining two parallel clipping planes that are arbitraily close to each other, one can obtain a tomographic slice through the significant surfaces.

A second data set that we analyzed is shown in Fig. 10. The data consist of cross sections of the magnitude of a 3-D vorticity vector obtained from a computer simulation. The transitions in the data are rendered as the surfaces in Fig. 11.

6. DISCUSSION AND CONCLUSIONS

In this paper we have presented an overview, as well as some specific techniques, for the analysis and display of 3-D data. Several of the problems and issues in the extraction of significant features, in the 3-D matching of contours, and in the use of computer graphics modeling for real experiment data have been considered jointly.

Note that some simple models were used in order to integrate the analysis of data with the computer graphics display techniques. For instance, we have described here the contours by B splines. In a related paper we have considered other piecewise polynomial or Gaussian steps for waveform and contour approximations.[27]

Another more serious limitation is that we have chosen to extract contours first and then to perform the contour-to-contour matching. An alternative, which is computationally much more complex, is to attempt matching from a fuller description of each cross section, such as from optical-flow information. We are currently studying such an approach that may provide a more reliable detailed match before contours are extracted and surfaces are constructed.

Among the other important issues that we did not address are the description and matching of small or incomplete features or contours. One useful approach for such a case is to combine the construction of significant surfaces with the display of unstructured data. If we refer, for instance, to Fig. 2, some of the smaller features detected within the brain cavity can be manipulated on a point-by-point basis in three dimensions, together with the significant surfaces, so as to capture and display 3-D relationships as well as details.

It is evident that a number of interesting and unresolved problems exist at all the stages of the overall process, either for feature extraction and characterization or for the rendering and display of different characteristics. Certainly, the computational requirements for the class of problems that we have addressed are large and will benefit greatly from the advances in architectures and algorithms, principally in the area of specialized computer graphics processors, that are now occurring. It is our view that such advances will greatly

broaden the range of applications of 3-D imaging, beyond the medical use to which they have been confined, to many other fields, principally in research.

ACKNOWLEDGMENTS

We thank W. Kollman for providing the three-dimensional vorticity data that were used for an example. The data are the results of modeling and simulation studies carried out at the Sandia Laboratories in Livermore on a Cray I supercomputer. This research was supported by IBM; Lockheed; TRW-ESL, and the Micro Electronics and Computer Science Research Opportunities program, University of California.

REFERENCES

1. W. A. Gaunt and P. N. Gaunt, *Three-Dimensional Reconstruction in Biology* (University Park, Baltimore, Md., 1978).
2. G. T. Herman, *Image Reconstruction from Projections* (Academic, New York, 1980).
3. D. A. Agard, "Optical sectioning microscopy: cellular architecture in three dimensions," Annu. Rev. Biophys. Bioeng. **13,** 191–219 (1984).
4. J. F. Claerbout, *Imaging the Earth Interior* (Blackwell, Oxford, 1985).
5. A. Macovski, *Medical Imaging Systems* (Prentice-Hall, Englewood Cliffs, N.J., 1983), Chap. 9.
6. E. J. Farrell, "Visual interpretation of complex data," IBM Syst. J. **26,** 174–200 (1987).
7. E. J. Farrell, "Color display and interactive interpretation of three-dimensional data," IMB J. Res. Develop. **27,** 356–366 (1983).
8. K. H. Hoehne, R. L. Delapaz, R. Bernstein, and R. C. Taylor, "Combined surface display and reformatting for the three-dimensional analysis of tomographic data," Invest. Radiol. **22,** 658–664 (1987).
9. F. Macias-Garza, A. C. Bovik, K. R. Diller, S. J. Aggarwal, and J. K. Aggarwal, "Digital reconstruction of three dimensional sectioned optical images," IEEE Trans. Acoust. Speech Signal Process. **ASSP-36,** 1067–1075 (1988).
10. R. A. Young, "The Gaussian derivative model for machine vision," J. Opt. Soc. Am. A **2** (13), p 39 (1985).
11. D. Marr and E. C. Hildreth, "The theory of edge detection," Proc. R. Soc. London Ser. B **207,** 187–217 (1980).
12. W. L. G. van Warmerdam and V. R. Algazi, "Image edge detection and width estimation with $\nabla^2 G$ filters," in *Intelligent Robots and Computer Vision,* D. P. Casasent, ed., Proc. Soc. Photo-Opt. Instrum. Eng. **848,** 102–107 (1987).
13. W. L. G. van Warmerdam, "An algorithm for generating a local description of the intensity transitions in an image," doctoral dissertation (Department of Electrical Engineering and Computer Science, University of California, Davis, Davis, Calif., 1988).
14. J. Canny, "A computational approach to edge detection," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-8,** 679–698 (1986).
15. F. Bergholm, "Edge focusing," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-9,** 726–741 (1987).
16. W. L. G. van Warmerdam and V. R. Algazi, "Describing 1-D intensity transitions with Gaussian derivatives at the resolutions matching their widths," IEEE Trans. Pattern Anal. Mach. Intell. (to be published).
17. A. F. Korn, "Toward a symbolic representation of intensity changes in images," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-10,** 610–625 (1988).
18. A. P. Witkin, "Space scale filtering," in *Proceedings of the Eighth International Joint Conference on Artificial Intelligence,* A. Bundy, ed. (Morgan Kaufmann, Los Altos, Calif., 1983), pp. 1019–1022.
19. R. H. Bartels, J. C. Beatty, and B. A. Barsky, *An Introduction to Splines for use in Computer Graphics and Geometric Modeling* (Morgan Kaufmann, New York, 1987).
20. B. A. Barsky and D. P. Greenberg, "Determining a set of B-spline control vertices to generate an interpolating surface," Comput. Graphics Image Process. **14,** 203–226 (1980).
21. B. A. Barsky, "End conditions and boundary conditions for uniform B-spline curve and surface representations," Comput. Industry **3,** 17–29 (1982).
22. D. F. Rogers, *Procedural Elements for Computer Graphics* (McGraw-Hill, New York, 1985).
23. *Starbase Graphics Techniques* (Hewlett-Packard, Fort Collins, Colo., 1987).
24. D. R. Warn, "Lighting controls for synthetic images," Comput. Graphics **17,** 13–21 (1983).
25. B. Phong, "Illumination for computer-generated pictures," Commun. ACM **18,** 311–317 (1975).
26. H. Gouraud, "Continuous shading of curved surfaces," IEEE Trans. Comput. **C-20,** 623–628 (1971).
27. W. L. G. van Warmerdam, C. C. Liu, and V. R. Algazi, "Space scale algorithms for segmenting and describing waveforms and contours," to be submitted to Comput. Graphics Image Process.